An Upper Bound for Aggregating Algorithm for Regression with Changing Dependencies

Yuri Kalnishkan

Computer Learning Research Centre and Department of Computer Science,
Royal Holloway, University of London, Egham, Surrey, TW20 0EX, UK
Yuri.Kalnishkan@rhul.ac.uk

Abstract. The paper presents a competitive prediction-style upper bound on the square loss of the Aggregating Algorithm for Regression with Changing Dependencies in the linear case. The algorithm is able to compete with a sequence of linear predictors provided the sum of squared Euclidean norms of differences of regression coefficient vectors grows at a sublinear rate.

1 Introduction

We consider the on-line learning scenario with signals. The following events are repeated for $t = 1, 2, \ldots$. The learner sequentially reads a signal $x_t \in \mathbb{R}^n$, makes a prediction $\gamma_t \in \mathbb{R}$ on the basis of the signal and past observations, and sees the true outcome $y_t \in [-Y, Y]$. The quality of the learner's predictions are assessed using a loss function $\lambda(\gamma, y)$, which is $(\gamma - y)^2$ in this paper.

We want to develop strategies for the learner making sure it suffers low cumulative loss $\mathrm{Loss}(T) = \sum_{t=1}^{T} \lambda(\gamma_t, y_t)$ over T steps. We approach this task within the competitive on-line prediction framework. According to this framework, no mechanism (probabilistic or other) generating the signals and outcomes is postulated. Instead we take a pool of prediction strategies and aim to build one that suffers loss not much worse than any strategy from the pool on every possible sequence of signals and outcomes.

In [Vov01] and [AW01] a prediction strategy is built that competes against the pool of all linear predictors outputting $\gamma_t = \theta' x_t$ for a fixed $\theta \in \mathbb{R}^n$. (Unless otherwise stated, all vectors in this paper are column vectors and M' is the transpose of a matrix or vector M.) The strategy called Aggregating Algorithm for Regression (AAR; also known as Vovk-Azoury-Warmuth predictor) suffers loss satisfying

$$\mathrm{Loss}_{\mathrm{AAR}}(T) \le \inf_{\theta \in \mathbb{R}^n} \left((\theta' x_t - y_t)^2 + a\|\theta\|^2 \right) + nY^2 \ln\left(\frac{TB^2}{an} + 1 \right) \qquad (1)$$

on every sequence $(x_1, y_1), (x_2, y_2), \ldots, (x_T, y_T)$, where $B = \max_{t=1,2,\ldots,T} \|x_t\|$ and $Y = \max_{t=1,2,\ldots,T} |y_t|$, $T = 1, 2, \ldots$, and the number $a > 0$ is the parameter of the strategy. (In this paper, $\|x\|$ denotes the Euclidean norm.) AAR does not need to know either B, Y, or the time horizon T from the start.

Intuitively, AAR covers the situation when we need to learn the 'right' θ on the fly, while making predictions. The extra term $nY^2 \ln\left(\frac{TB^2}{an}+1\right)$ grows logarithmically in T, which is a very small price to pay for not knowing the 'right' θ from the start. One may want to generalise the result to the situation when θ changes with time. Consider a prediction strategy using a sequence $\theta_1, \theta_2, \ldots$ to predict in the following way. On step t it predicts $\gamma_t = u_t' x_t$, where $u_t = \sum_{i=1}^{t} \theta_i$. Clearly, aiming to do as well as *any* such sequence is hopeless. To every sequence $(x_1, y_1), (x_2, y_2), \ldots$ one can fit a sequence $u_1, u_2, \ldots$ suffering zero loss, provided $x_t \neq 0$. However, one can hope to compete with a sequence of slowly changing u_t. If $\sum_{t=1}^{T} \|\theta_t\| = \|\theta_1\|^2 + \sum_{t=2}^{T} \|u_t - u_{t-1}\|^2$ grows slowly, can we have a strategy with an upper bound on the loss similar to (1)?

This problem has been approached using a variety of techniques. In [HW01] an algorithm based on Bregman divergence and gradient descent-type methods was proposed. The bounds obtained in [HW01] have multiplicative constants in front of the competitors' losses. In [BK07a] an algorithm called Aggregating Algorithm for Regression with Changing Dependencies (AARCh) based on Vovk's Aggregating Algorithm and extending the construction of AAR from [Vov01] was proposed. The bounds form [BK07a] have no multiplicative constant, but the final result is rather week. A recent paper [MVC15] has proposed a strategy LASER based on the last-step min-max approach of [For99]. The strategy takes a function $v(T) = O(T)$ and $a > 0$ as parameters and suffers loss satisfying

$$\mathrm{Loss}_{\mathrm{LASER}}(T) \leq \inf_{\substack{u_1, u_2, \ldots, u_T \in \mathbb{R}^n: \\ \sum_{t=2}^{T} \|u_t - u_{t-1}\|^2 \leq v(T)}} \left(\sum_{t=1}^{T} (u_t' x_t - y_t)^2 + a\|u_1\|^2 \right) + nY^2 \ln\left(\frac{TB^2}{an}+1\right) + O((v(T))^{1/3} T^{2/3}) \ . \quad (2)$$

The bound is far superior to that from [BK07a].

In this paper we improve the upper bound for AARCh from [BK07a] to achieve an extra term $O((v(T))^{1/3}T^{2/3})$ matching that of (2). The multiplicative constant in the extra term exhibits better dependency on the dimension, $n^{1/3}$ instead of $n^{2/3}$ in [MVC15].

As with LASER, in order to achieve this, AARCh should be optimised from the start using the prior knowledge of the time horizon T, the value of $v(T)$, B, and Y. Applying the Aggregating Algorithm allows one to dispense with some prior knowledge (with a notable exception exception of Y) but complicates the strategy.

One may note that the problem of competing with a sequence of us can be thought of as an extension to the regression framework of the problem of competing against switching experts in prediction with expert advice; see [AKCV12] for a comparison of bounds given by different approaches.

2 Preliminaries

2.1 Games and Prediction Strategies

A *game* $\mathfrak{G}$ is a triple of an *outcome space* Ω, prediction space Γ and a *loss function* $\lambda : \Gamma \times \Omega \to [0, +\infty]$.

A *prediction strategy* $\mathcal{S}$ for a game $\mathfrak{G}$ working with signals from a *signal space* X is a mapping $\mathcal{S} : (X \times \Omega)^* \times X \to \Gamma$. Intuitively, $\mathcal{S}$ supplies predictions for the learner acting according to this protocol:

Protocol 1.
(1) FOR $t = 1, 2, \ldots$
(2) the learner reads signal $x_t \in X$
(3) the learner produces $\gamma_t \in \Gamma$
(4) the learner sees $y_t \in \Omega$
(5) END FOR

On a sequence $(x_1, y_1), (x_2, y_2), \ldots, (x_T, y_T)$ the learner using the strategy $\mathcal{S}$ suffers cumulative loss

$$\mathrm{Loss}_{\mathcal{S}}(T) = \sum_{t=1}^{T} \lambda(\gamma_t, y_t) = \sum_{t=1}^{T} \lambda(\mathcal{S}(x_1, y_1, \ldots, x_{t-1}, y_{t-1}, x_t), y_t) \ .$$

The index $\mathcal{S}$ will be dropped if it is clear from the context.

We will be considering square-loss games with $\Omega \subseteq \mathbb{R}$, $\Gamma = \mathbb{R}$ and $\lambda(\gamma, y) = (\gamma - y)^2$. For Ω we take different subsets of $\mathbb{R}$. Strictly speaking the theory of the Aggregating Algorithm (see [Vov01]) applies to the bounded game with $\Omega = [-Y, Y]$. However it often happens that the algorithm does not need to know Y in advance and Y only appears in the bound. Then we can say the algorithm applies to the case $\Omega = \mathbb{R}$.

2.2 Aggregating Algorithm for Regression with Changing Dependencies

The Aggregating Algorithm for Regression with Changing Dependencies (AARCh) was introduced in [BK07a] (see also [BK07b] for numerical experiments).

AARCh is a prediction strategy for a game with real outcomes and predictions and signals from $\mathbb{R}^n$. It takes as parameters a sequence $a_1, a_2, \ldots > 0$ and on step T predicts $\gamma_T = \tilde{y}'(\bar{K} + I)^{-1}\bar{k}$, where

$$\tilde{y} = \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_{T-1} \\ 0 \end{pmatrix} \ , \qquad \bar{k} = \begin{pmatrix} \frac{1}{a_1} x_1' x_T \\ \left(\frac{1}{a_1} + \frac{1}{a_2}\right) x_2' x_T \\ \vdots \\ \left(\frac{1}{a_1} + \frac{1}{a_2} + \cdots + \frac{1}{a_T}\right) x_T' x_T \end{pmatrix} \ ,$$

and

$$\bar{K} = \begin{pmatrix} \frac{1}{a_1} x_1' x_1 & \frac{1}{a_1} x_1' x_2 & \dots & \frac{1}{a_1} x_1' x_T \\ \frac{1}{a_1} x_2' x_1 & \left(\frac{1}{a_1} + \frac{1}{a_2}\right) x_2' x_2 & \dots & \left(\frac{1}{a_1} + \frac{1}{a_2}\right) x_2' x_T \\ \vdots & \vdots & \ddots & \vdots \\ \frac{1}{a_1} x_T' x_1 & \left(\frac{1}{a_1} + \frac{1}{a_2}\right) x_T' x_2 & \dots & \left(\frac{1}{a_1} + \dots + \frac{1}{a_T}\right) x_T' x_T \end{pmatrix}$$

(this is the dual form given in Section 3.3 of [BK07a]).

The algorithm is obtained by applying the Aggregating Algorithm in the bounded square loss game to a particular set of experts.

3 Main Result

In this section we formulate and discuss upper bounds on the cumulative square loss of AARCh.

Theorem 1. *For every sequence $(x_1, y_1), (x_2, y_2), \dots, (x_T, y_T) \in \mathbb{R}^n \times \mathbb{R}$, the square loss of the learner using AARCh with positive parameters $a_1, a, \dots, a$ satisfies*

$$\mathrm{Loss}(T) \le \inf_{u_1,\dots,u_T \in \mathbb{R}^n} \left(\sum_{t=1}^{T} (u_t' x_t - y_t)^2 + a_1 \|u_1\|^2 + a \sum_{t=2}^{T} \|u_t - u_{t-1}\|^2 \right) + nY^2 \ln\left(1 + \frac{TB^2}{a_1 n}\right) + Y^2 B \left(T - \frac{1}{2}\right) \sqrt{\frac{n}{a}} - nY^2 \ln 2 + \alpha(T, a) \ , \quad (3)$$

where $Y = \max_{t=1,\dots,T} |y_t|$, $B = \max_{t=1,\dots,T} \|x_t\|$, and

$$\alpha(T, a) = nY^2 \left(1 + \frac{B^2}{2an} - \sqrt{\frac{B^4}{4a^2 n^2} + \frac{B^2}{an}}\right)^{2(T-1)} \le nY^2 \ . \quad (4)$$

Clearly, the bound only makes sense if the terms on the right, apart from $\sum_{t=1}^{T} (u_t' x_t - y_t)^2$, are not too large. If the outcomes y_t are bounded, $|y_t| \le Y$, then it is too easy to get loss not exceeding $Y^2 T$ by predicting 0 consistently. Thus an extra term growing faster than $O(T)$ makes little sense and $O(T)$ can only be useful if the constant is small. On the other hand, competing with sequences of u_t such that $\|u_t - u_{t-1}\|$ is large is futile: as pointed out in [MVC15], the sequence $u_t = x_t y_t / \|x_t\|^2$ leads to zero loss as long as $x_t \neq 0$. Thus one may want to obtain an extra term of the order $O(T)$ and, if possible, $o(T)$, by restricting the variability of u_t.

Let us find a optimising the sum $a \cdot \sum_{t=2}^{T} \|u_t - u_{t-1}\|^2 + Y^2 B (T - 1/2) \sqrt{\frac{n}{a}}$. If Y, B, T, and the order $v(T)$ of the growth of the sum $\sum_{t=2}^{T} \|u_t - u_{t-1}\|^2$ (cf. $V^{(2)}$ in [MVC15]) are known in advance, we can find the optimal a as follows.

Lemma 1. *For all positive v and c the minimum $\min_{a>0} \left(av + \frac{c}{\sqrt{a}}\right)$ is achieved at $a = \left(\frac{c}{2v}\right)^{2/3}$ and equals $\frac{3}{2^{2/3}} c^{2/3} v^{1/3}$.*

Proof. As $a \to 0$ or $a \to +\infty$, the expression tends to $+\infty$. We get the minimum by equating to zero the derivative

$$\frac{\partial}{\partial a}\left(av + \frac{c}{\sqrt{a}}\right) = v - \frac{c}{2a^{3/2}} \ .$$

□

Corollary 1. *For every function $v : \{2, 3, \ldots\} \to (0, +\infty)$ and every sequence $(x_1, y_1), (x_2, y_2), \ldots, (x_T, y_T) \in \mathbb{R}^n \times \mathbb{R}$ such that $\|x_t\| \le B$ and $|y_t| \le Y$ for $t = 1, 2, \ldots, T$, the square loss of the learner using AARCh with parameters $a_1, a, \ldots, a$, where $a_1 > 0$ and*

$$a = a(T) = \frac{Y^{4/3}B^{2/3}n^{1/3}}{2^{2/3}} \cdot \frac{(T - 1/2)^{2/3}}{(v(T))^{2/3}} \ , \tag{5}$$

satisfies

$$\begin{aligned}\mathrm{Loss}(T) \le \inf_{\substack{u_1,\ldots,u_T \in \mathbb{R}^n \\ \sum_{t=2}^T \|u_t - u_{t-1}\|^2 \le v(T)}} &\left(\sum_{t=1}^T (u_t' x_t - y_t)^2 + a_1\|u_1\|^2\right) \\ &+ nY^2 \ln\left(\frac{TB^2}{a_1 n} + 1\right) + \frac{3}{2^{2/3}} Y^{4/3}B^{2/3}n^{1/3}\left(T - \frac{1}{2}\right)^{2/3}(v(T))^{1/3} \\ &- nY^2 \ln 2 + \alpha(T, a(T)), \end{aligned} \tag{6}$$

where $\alpha(T, a(T)) \le nY^2$ is given by (4).

If, moreover, $v(t) = o(T)$ and $1/v(T) = o(T^2)$ as $T \to +\infty$, then

$$\alpha(T, a(T)) \le nY^2 e^{-2\frac{B(T-1)}{\sqrt{a(T)n}}\left(1 - \frac{B}{2\sqrt{a(T)n}}\right)} \to 0 \tag{7}$$

as $T \to +\infty$.

Proof. It is easy to see that

$$0 < 1 + \frac{b}{2} - \sqrt{\frac{b^2}{4} + b} \le 1 + \frac{b}{2} - \sqrt{b}$$

for all $b \ge 0$. Applying the inequality $\ln(1 + x) \le x$ yields upper bound (7).

Since $v(t) = o(T)$, we get $(T - 1/2)/v(T) \to +\infty$ and thus $a(T) \to +\infty$ as $T \to +\infty$. The condition $1/v(T) = o(T^2)$ implies $T/\sqrt{a} \to +\infty$. Therefore the power in the term on the right-hand side tends to $-\infty$ and the term itself tends to 0 as $T \to +\infty$. □

The main component of the extra term in the bound has the same order of growth in T, namely, $T^{2/3}(v(T))^{1/3}$, as in the bound for LASER in Corollary 12 of [MVC15]. If $v(T) = o(T)$ as $T \to +\infty$, the order of growth is sublinear.

However, the multiplicative coefficient differs and we get $\frac{3}{2^{2/3}} Y^{4/3} B^{2/3} n^{1/3}$ instead of $3 \cdot 2^{1/3} Y^{4/3} B^{2/3} n^{2/3}$. Our term is smaller by the factor of $2n^{1/3}$. See Remark 2 below for a discussion of the power[1] of n.

Having to know the time horizon T in advance to choose a is annoying. This problem can be eliminated by applying the Aggregating Algorithm. Suppose we know Y, B, and $v(T)$. Then we can apply the Aggregating Algorithm to a countable number of instances of AARCh, each using a from (5), $T = 2, 3, \ldots$ Let us assign to the instance corresponding to T a prior $p_0(T) = \frac{6}{\pi^2 (T-1)^2}$, $T = 2, 3, \ldots$ and apply the AA. Bound (10) with $\eta = 1/(2Y^2)$ and $C(\eta) = 1$ give us the following corollary.

Corollary 2. *For $Y > 0$, $B > 0$, $a_1 > 0$ and a function $v : \{2, 3, \ldots\} \to (0, +\infty)$ there is a prediction strategy $\mathcal{S}$ that on every sequence $(x_1, y_1), \ldots, (x_T, y_T) \in \mathbb{R}^n \times \mathbb{R}$ such that $\|x_t\| \le B$ and $|y_t| \le Y$ for all $t = 1, 2, \ldots, T$ suffers square loss*

$$\mathrm{Loss}_{\mathcal{S}}(T) \le \inf_{\substack{u_1, \ldots, u_T \in \mathbb{R}^n \\ \sum_{t=2}^T \|u_t - u_{t-1}\|^2 \le v(T)}} \left(\sum_{t=1}^T (u_t' x_t - y_t)^2 + a_1 \|u_1\|^2 \right) + nY^2 \ln \left(\frac{TB^2}{a_1 n} + 1 \right) + \frac{3}{2^{2/3}} Y^{4/3} B^{2/3} n^{1/3} \left(T - \frac{1}{2} \right)^{2/3} (v(T))^{1/3} + 2Y^2 \ln T + 2Y^2 \ln \frac{\pi^2}{6} - nY^2 \ln 2 + \alpha_{Y,B,v}(T) \ , \quad (8)$$

where $\alpha_{Y,B,v}(T) \le nY^2$ and tends to zero as $T \to +\infty$ provided $v(T) = o(T)$ and $1/v(T) = o(T^2)$.

While the Aggregating Algorithm provides a way of computing $\mathcal{S}$, the procedure is complicated. Arguing in a similar way, we can eliminate the dependency on B and reduce the dependency on the order of growth of $v(t)$ at a price of making the strategy even more complicated. The dependency on Y cannot be overcome this way though as the Aggregating Algorithm assumes Y is finite and known. (As Y grows to infinity, the maximum value $\eta = 1/(2Y^2)$ such that the game is mixable vanishes and renders bound (10) useless.)

In the rest of the paper we prove Theorem 1. Section 4 covers the steps done in [BK07a], Section 5 presents the original material, and Section 6 contains some remarks on the proof.

4 Deriving the Upper Bound on AARCh

In this section we review the derivation of the upper bound on AARCh from [BK07a] starting with the basics of prediction with expert advice and Vovk's Aggregating Algorithm after [Vov98,Vov01].

[1] The fact that the powers of n and T sum to 1 makes the straightforward kernelisation of the bound based on the representer theorem useless. This observation may potentially lead to a lower bound.

4.1 Prediction with Expert Advice

The goal of prediction with expert advice is constructing prediction strategies competitive with other strategies from a pool can be addressed within the framework of prediction with expert advice.

Suppose we have a *pool of experts* Θ. Predictions output by experts at any moment in time can be described by a function $\Theta \to \Gamma$. Let $\mathcal{E} \subseteq \Gamma^\Theta$ be a set of such functions that we allow (e.g., measurable functions). Prediction with expert advice is concerned with building merging strategies $\mathcal{M} : (\mathcal{E} \times \Omega)^* \times \mathcal{E} \to \Gamma$. Intuitively, $\mathcal{M}$ supplies predictions for the learner acting according to this protocol:

Protocol 2.
(1) FOR $t = 1, 2, \ldots$
(2) the learner reads experts' predictions γ_t^θ, $\theta \in \Theta$
(3) the learner produces $\gamma_t \in \Gamma$
(4) the learner sees $y_t \in \Omega$
(5) END FOR

Over T steps expert θ suffers loss $\mathrm{Loss}_\theta(T) = \sum_{t=1}^T \lambda(\gamma_t^\theta, y_t)$. Prediction with expert advice looks for merging strategies making sure that the cumulative loss of the learner is not much greater than the loss of every expert $\theta \in \Theta$.

4.2 Aggregating Algorithm

The Aggregating Algorithm (AA) was proposed in [Vov90,Vov98]. It is a rather general merging strategy.

The Aggregating Algorithm takes as parameters $\eta > 0$, a (prior) distribution P_0 on Θ, and a substitution rule, which will be defined later. On step t it forms the *generalised prediction*, which is a function $g_t : \Omega \to [0, +\infty]$ given by

$$g_t(y) = -\frac{1}{\eta} \ln \frac{\int_\Theta e^{-\eta\lambda(\gamma_t^\theta, y)} e^{-\eta \,\mathrm{Loss}_\theta(t-1)} P_0(d\theta)}{\int_\Theta e^{-\eta \,\mathrm{Loss}_\theta(t-1)} P_0(d\theta)} \quad .$$

The generalised prediction is then converted to a prediction γ_t such that $\lambda(\gamma_t, y) \le C(\eta) g_t(y)$ for all $y \in \Omega$. Here $C(\eta)$ is the minimum constant permitted for the game. It is shown in Section 2.4 of [Vov01] that for the bounded square-loss game with $\Omega = [-Y, Y]$ we can take $C(\eta) = 1$ for $\eta \le 1/(2Y^2)$ (as can be seen from (10) below, in such situations one wants to maximise η, so $\eta = 1/(2Y^2)$ is used). A *substitution rule* maps generalised predictions into predictions. A convenient substitution rule leads to simple algorithms.

The Aggregating Algorithm ensures that the learner's loss satisfies

$$\mathrm{Loss}_{\mathrm{AA}}(T) \le -\frac{C(\eta)}{\eta} \ln \int_\Theta e^{-\eta \,\mathrm{Loss}_\theta(T)} P_0(d\theta) \tag{9}$$

(this can be checked by induction). This inequality holds for all possible sequences of outcomes. If the pool is finite or countable, the integral reduces to

the sum and by dropping from the sum all terms except for one we obtain the inequality

$$\mathrm{Loss}_{\mathrm{AA}}(T) \le C(\eta)\,\mathrm{Loss}_\theta(T) + \frac{C(\eta)}{\eta}\ln\frac{1}{P_0(\theta)} \tag{10}$$

for every expert θ. If the pool is not countable, as it is below, this general trick does not apply and we need to upper bound (9) for the particular case.

4.3 Constructing the Bound for AARCh

AARCh is obtained by applying AA in the context of a bounded square-loss game with the outcome space $\Omega = [-Y, Y]$ and the signal space $X = \mathbb{R}^n$ to the following experts. Fix a positive integer T and let $\Theta = (\mathbb{R}^n)^T$. We can consider elements of Θ as vectors of nT real components or sequences of T vectors from $\mathbb{R}^n$, $\theta = (\theta_1, \theta_2, \dots, \theta_T)$. On step t expert θ predicts $\gamma_t^\theta = (\sum_{i=1}^t \theta_i)' x_t$.

Take $\eta = 1/(2Y^2)$; as mentioned above, we get $C(\eta) = 1$ for the bounded square-loss game. On Θ we consider the Gaussian prior with the density

$$p_0(\theta) = \prod_{t=1}^T \left[\left(\frac{\eta a_t}{\pi}\right)^{n/2} e^{-\eta a_t \|\theta_t\|^2}\right] = \left(\prod_{t=1}^T a_t^{n/2}\right)\left(\frac{\eta}{\pi}\right)^{Tn/2} e^{-\eta\sum_{t=1}^T a_t\|\theta_t\|^2} \ ,$$

where $a_1, a_2, \dots, a_T > 0$ are the parameters of AARCh.

We will omit the derivation of the formulas for AARCh given in Section 2.2, but give the derivation of the upper bound. Bound (9) ensures that

$$\mathrm{Loss}_{\mathrm{AARCh}}(T) \le -\frac{1}{\eta}\ln\int_{\mathbb{R}^{nT}} e^{-\eta\,\mathrm{Loss}_\theta(T)} p_0(\theta)\,d\theta \ . \tag{11}$$

The loss of expert θ equals

$$\mathrm{Loss}_\theta(T) = \sum_{t=1}^T \left(\left(\sum_{i=1}^t \theta_i\right)' x_t - y_t\right)^2 = \sum_{t=1}^T (\theta' w_t - y_t)^2 \ ,$$

where θ is interpreted as a column vector and

$$w_t' = (\underbrace{x_t', \dots, x_t'}_{t \text{ times}}, \underbrace{0, \dots, 0}_{(T-t)n \text{ zeros}})' \ .$$

This is a quadratic form in θ. Multiplying $e^{-\eta\,\mathrm{Loss}_\theta(T)}$ by $p_0(\theta)$ adds a quadratic term to the power. The integral can be evaluated using the following proposition.

Proposition 1. *For a quadratic form $Q(\theta)$, $\theta \in \mathbb{R}^m$, with the quadratic part $\theta' A\theta$, where A is a symmetric positive definite $(m \times m)$-matrix, we get*

$$\int_{\mathbb{R}^m} e^{-Q(\theta)} = e^{-Q_0}\frac{\pi^{m/2}}{\sqrt{\det A}} \ ,$$

where $Q_0 = \min_{\theta\in\mathbb{R}^m} Q(\theta)$.

The proof of the proposition is essentially by completing the square and integration by substitution.

The matrix of the quadratic part of the negation of the form in the power in (11) is

$$\eta A = \eta \sum_{t=1}^{T} w_t w_t' + \eta \begin{pmatrix} a_1 I & & 0 \\ & \ddots & \\ 0 & & a_T I \end{pmatrix} \ .$$

It is easy to see that A is positive definite.

Proposition 2.

$$\mathrm{Loss}_{\mathrm{AARCh}}(T) \le \inf_{\theta_1,\dots,\theta_T \in \mathbb{R}^n} \left(\mathrm{Loss}_{\theta_1,\dots,\theta_T}(T) + \sum_{t=1}^{T} a_t \|\theta_t\|^2 \right) + Y^2 \ln \frac{\det A}{\prod_{t=1}^{T} a_t^n} \ ,$$

where

$$A = \begin{pmatrix} \sum_{t=1}^{T} x_t x_t' + a_1 I & \sum_{t=2}^{T} x_t x_t' & \sum_{t=3}^{T} x_t x_t' & \vdots & x_T x_T' \\ \sum_{t=2}^{T} x_t x_t' & \sum_{t=2}^{T} x_t x_t' + a_2 I & \sum_{t=3}^{T} x_t x_t' & \vdots & x_T x_T' \\ \sum_{t=3}^{T} x_t x_t' & \sum_{t=3}^{T} x_t x_t' & \sum_{t=3}^{T} x_t x_t' + a_3 I & \vdots & x_T x_T' \\ \cdots & \cdots & \cdots & \ddots & \vdots \\ x_T x_T' & x_T x_T' & x_T x_T' & \cdots & x_T x_T' + a_T I \end{pmatrix} \ .$$

It remains to upper bound the determinant of A.

5 Upper Bounding the Determinant

By Theorem 7 of Section 2.10, [BB61], the determinant of a positive definite matrix does not exceed the product of determinants of the minors. Hence

$$\det A \le \det \left(\sum_{t=1}^{T} x_t x_t' + a_1 I \right) \det A_2 \ , \tag{12}$$

where

$$A_2 = \begin{pmatrix} \sum_{t=2}^{T} x_t x_t' + a_2 I & \sum_{t=3}^{T} x_t x_t' & \vdots & x_T x_T' \\ \sum_{t=3}^{T} x_t x_t' & \sum_{t=3}^{T} x_t x_t' + a_3 I & \vdots & x_T x_T' \\ \cdots & \cdots & \ddots & \vdots \\ x_T x_T' & x_T x_T' & \cdots & x_T x_T' + a_T I \end{pmatrix} \ .$$

Proposition 3 ([CBCG05]). *For every positive integer T, all vectors $x_1, x_2, \dots, x_T \in \mathbb{R}^n$ such that $\|x_t\| \le B$, $t = 1, 2, \dots, T$, and all $a > 0$ we have*

$$\frac{1}{a^n} \det \left(\sum_{t=1}^{T} x_t x_t' + aI \right) \le \left(\frac{TB^2}{an} + 1 \right)^n \ .$$

The proof is by Proposition 5 given below.

We will now simplify the structure of A_2. From every block row, except for the last, we subtract the next row. We start from the first row and do this from top to bottom. Then from every block column, except for the last, we subtract the next block column, going right to left. This results in a block tridiagonal matrix $\widetilde{A}_2$ given by

$$\begin{pmatrix} x_2x_2'+(a_2+a_3)I & -a_3I & & 0 & \\ -a_3I & x_3x_3'+(a_3+a_4)I & & -a_4I & \\ & \ddots & \ddots & & \ddots \\ & & -a_{T-1}I & x_{T-1}x_{T-1}'+(a_{T-1}+a_T)I & -a_TI \\ & & 0 & -a_TI & x_Tx_T'+a_TI \end{pmatrix}$$

Subtracting block row j from block row i amounts to multiplication on the left by a block elementary matrix L_{ij} with determinant 1. Subtracting block column j from block row i amounts to multiplication on the right by $L'_{i,j}$. Thus

$$\widetilde{A}_2 = L_{T-1,T}L_{T-2,T-1}\cdots L_{1,2}A_2L'_{1,2}\cdots L'_{T-2,T-1}L'_{T-1,T}$$

and therefore $\widetilde{A}_2$ is still symmetric and positive definite and $\det \widetilde{A}_2 = \det A_2$.

We now set $a_2 = a_3 = \ldots = a_T = a$ and let

$$\overline{A}_2 = \frac{1}{a}\widetilde{A}_2 = \begin{pmatrix} \frac{x_2x_2'}{a}+2I & -I & & 0 & \\ -I & \frac{x_3x_3'}{a}+2I & & -I & \\ & \ddots & \ddots & & \ddots \\ & & -I & \frac{x_{T-1}x_{T-1}'}{a}+2I & -I \\ & & 0 & -I & \frac{x_Tx_T'}{a}+I \end{pmatrix}$$

The determinant of a block tridiagonal matrix can be calculated as follows.

Proposition 4 ([Sal06]). *The determinant of a block tridiagonal matrix*

$$M = \begin{pmatrix} G_1 & E_2 & 0 & & \\ F_2 & G_2 & E_3 & & \\ & \ddots & \ddots & \ddots & \\ & & F_{m-1} & G_{m-1} & E_m \\ & & 0 & F_m & G_m \end{pmatrix}$$

is $\det M = \prod_{k=1}^{m} \det \Lambda_k$, *where* $\Lambda_1 = G_1$ *and* $\Lambda_k = G_K - F_k\Lambda_{k-1}^{-1}E_k$, $k = 2, 3, \ldots, m$, *provided all required inversions can be performed.*

Proof. The proof is by reducing the matrix to the block upper triangular form and taking the product of determinants of the diagonal blocks. By subtracting from the second block row the first block row multiplied on the left by $F_2G_1^{-1}$, we eliminate F_2 and get Λ_2 in place of G_2. The rest is by induction. □

We get

$$\det \overline{A}_2 = \prod_{t=2}^{T} \det \Lambda_t \ , \tag{13}$$

where $\Lambda_2 = x_2 x_2'/a + 2I$, $\Lambda_t = x_t x_t'/a + 2I - \Lambda_{t-1}^{-1}$ for $t = 3, \ldots, T-1$, and $\Lambda_T = x_T x_T'/a + I - \Lambda_{T-1}^{-1}$.

Lemma 2. *All Λ_t, $t = 2, \ldots, T$, are well-defined symmetric positive definite matrices.*

Proof. Let us prove by induction that, for $t = 2, 3, \ldots, T-1$, Λ_t is symmetric positive definite and all its eigenvalues are greater than or equal to 1. The eigenvalues of Λ_2 are $2 + \|x_t\|^2/a, 2, \ldots, 2$ so the base of the induction holds.

If Λ_{t-1} satisfies the induction hypothesis, then it is invertible, Λ_{t-1}^{-1} is symmetric positive definite and and all its eigenvalues are less than or equal to 1. The eigenvalues of $x_t x_t'/a + 2I$ are greater than or equal to 2. By the Courant-Fischer min-max theorem ([HJ13], Theorem 4.2.6)) the eigenvalues of $x_t x_t'/a + 2I - \Lambda_{t-1}^{-1}$ are greater than or equal to 1.

A similar argument implies that eigenvalues of Λ_T are non-negative. However, if it is singular, then (13) implies that $\det \overline{A}_2 = 0$. Since $\overline{A}_2$ is positive definite, Λ_T is non-singular. □

The matrix recursive formulas for Λ_t are difficult to analyse. We will use the following proposition to reduce them to scalar formulas.

Proposition 5. *If M is a symmetric positive semidefinite $(m \times m)$-matrix, then*

$$\det M \le \left(\frac{\operatorname{tr} M}{m} \right)^m \ ;$$

if M is positive definite, then

$$\operatorname{tr} M^{-1} \ge \frac{m^2}{\operatorname{tr} M} \ .$$

(Notation $\operatorname{tr} M$ is used for the *trace* of a matrix M.)

Proof. Let $\lambda_1, \lambda_2, \ldots, \lambda_m$ be the eigenvalues of M, counting multiplicities. The inequalities for the arithmetic, geometric, and harmonic means

$$(\lambda_1 \lambda_2 \ldots \lambda_m)^{1/m} \le \frac{\lambda_1 + \lambda_2 + \ldots + \lambda_m}{m} \ ,$$
$$\frac{m}{\frac{1}{\lambda_1} + \frac{1}{\lambda_2} + \ldots + \frac{1}{\lambda_m}} \le \frac{\lambda_1 + \lambda_2 + \ldots + \lambda_m}{m}$$

(see Section 1.16 of [BB61]) imply the proposition. □

Corollary 3. *The determinant of $\overline{A}_2$ satisfies $\det \overline{A}_2 \le (r_2 r_3 \ldots r_T)^n$, where the sequence r_t, $t = 2, 3, \ldots, T$, is defined by $r_2 = b + 2$, $r_t = b + 2 - 1/r_{t-1}$ for $t = 3, \ldots, T-1$, and $r_T = b + 1 - 1/r_{T-1}$ with $b = \frac{B^2}{an}$.*

Proof. It follows by induction that $\operatorname{tr} \Lambda_t/n \le r_t$. Indeed,

$$\begin{aligned}
\frac{\operatorname{tr} \Lambda_2}{n} &\le \frac{B^2}{an} + 2 = r_2 \ , \\
\frac{\operatorname{tr} \Lambda_t}{n} &\le \frac{B^2}{an} + 2 - \frac{\operatorname{tr} \Lambda_{t-1}^{-1}}{n} \le \frac{B^2}{na} + 2 - \frac{n}{\operatorname{tr} \Lambda_{t-1}} \\
&\le b + 2 - \frac{1}{r_{t-1}} = r_t \ , \quad t = 3, \dots, T-1,
\end{aligned}$$

and

$$\frac{\operatorname{tr} \Lambda_T}{n} \le \frac{B^2}{an} + 1 - \frac{\operatorname{tr} \Lambda_{T-1}^{-1}}{n} \le b + 1 - \frac{1}{r_{T-1}} = r_T \ .$$

We get $\det \Lambda_t \le r_t^n$ and the corollary follows by (13). □

The products $r_2 r_3 \dots r_t$ form a recurrent sequence, which is easy to analyse.

Lemma 3. *The determinant of $\overline{A}_2$ satisfies $\det\left(\overline{A}_2\right) \le (d_T - d_{T-1})^n$, where the sequence d_t, $t = 0, 1, 2, \dots$, is defined by $d_0 = 0$, $d_1 = 1$, and $d_t = (b+2)d_{t-1} - d_{t-2}$ for $t = 2, 3, \dots$ with $b = \frac{B^2}{an}$.*

Proof. By induction we get $d_t = r_2 \dots r_t$ for $t = 2, 3, \dots, T-1$ and $d_T = r_2 \dots r_{T-1}(r_T + 1) = r_2 \dots r_{T-1} r_T + d_{T-1}$. □

We need to study the behaviour of d_t.

Lemma 4. *For every $b > 0$ the sequence d_t from Lemma 3 satisfies*

$$d_T - d_{T-1} = \frac{1}{2}\left(\lambda_1^{T-1}\left(1 + \frac{b}{\sqrt{b^2+4b}}\right) + \lambda_2^{T-1}\left(1 - \frac{b}{\sqrt{b^2+4b}}\right)\right) \ ,$$

where $\lambda_1 = 1 + \frac{b}{2} + \frac{1}{2}\sqrt{b^2+4b}$, $\lambda_2 = 1 + \frac{b}{2} - \frac{1}{2}\sqrt{b^2+4b}$, and $T = 1, 2, \dots$

Proof (Sketch). The recurrent formula for d_t can be written in the matrix form as

$$\begin{pmatrix} d_t \\ d_{t-1} \end{pmatrix} = R \begin{pmatrix} d_{t-1} \\ d_{t-2} \end{pmatrix} \ , \text{ where } \ R = \begin{pmatrix} b+2 & -1 \\ 1 & 0 \end{pmatrix} \ ,$$

and thus

$$\begin{pmatrix} d_T \\ d_{T-1} \end{pmatrix} = R^{T-1} \begin{pmatrix} d_1 \\ d_0 \end{pmatrix} = R^{T-1} \begin{pmatrix} 1 \\ 0 \end{pmatrix} \ .$$

In order to calculate R^{T-1}, we need to represent R in a convenient form. One can check that λ_1 and λ_2 are the eigenvalues of R and the corresponding eigenvectors can be chosen as

$$\begin{aligned}
v_1 &= (-\sqrt{b} - \sqrt{b+4}, \sqrt{b} - \sqrt{b+4})' \ , \\
v_2 &= (-\sqrt{b} + \sqrt{b+4}, \sqrt{b} + \sqrt{b+4})' \ .
\end{aligned}$$

We get $R = V \Lambda V^{-1}$, where Λ is the diagonal matrix with diagonal elements λ_1 and λ_2 and the columns of V are v_1 and v_2. Raising to power $T-1$ can be done as $R^{T-1} = V \Lambda^{T-1} V^{-1}$. The lemma follows by direct calculation. □

The following simple facts will be used to upper bound $d_T - d_{T-1}$.

Lemma 5. *For every $b > 0$ we get*

$$\frac{b}{\sqrt{b^2+4b}} \leq \frac{\sqrt{b}}{2} \ .$$

For every $b \geq 0$ we get

$$\frac{\lambda_2}{\lambda_1} = \lambda_2^2 = \left(1 + \frac{b}{2} - \frac{1}{2}\sqrt{b^2+4b}\right)^2 \leq 1 \ ,$$

and

$$\ln \lambda_1 \leq \sqrt{b} \ ,$$

where λ_1 and λ_2 are from Lemma 4.

Proof (Sketch). The first inequality follows from

$$\frac{b}{\sqrt{b^2+4b}} = \frac{\sqrt{b}}{\sqrt{b+4}} \leq \frac{\sqrt{b}}{2} \ .$$

The equality involving lambdas can be checked by direct calculation. The inequality follows from

$$\frac{b}{2} \leq \sqrt{\frac{b^2}{4} + b} < 1 + \frac{b}{2} \ .$$

The last inequality follows by differentiation:

$$\frac{d}{db} \ln \lambda_1 = \frac{1}{\sqrt{b^2+4b}} \leq \frac{1}{\sqrt{4b}} = \frac{1}{2\sqrt{b}} = \frac{d}{db}\sqrt{b} \ ,$$

while for $b = 0$ we get $\ln \lambda_1 = \sqrt{b} = 0$. □

We can now upper bound the extra term in Proposition 2 as

$$Y^2 \ln \frac{\det A}{a_1^n a^{n\cdot(T-1)}} \leq nY^2 \ln\left(\frac{TB^2}{a_1 n} + 1\right) + Y^2 \ln \det \overline{A}_2 \ ,$$

where $Y^2 \ln \det \overline{A}_2 \leq nY^2 \ln(d_T - d_{T-1})$ and

$$\ln(d_T - d_{T-1}) \leq \ln \frac{1}{2}\left(\lambda_1^{T-1}\left(1 + \frac{\sqrt{b}}{2}\right) + \lambda_2^{T-1}\right) =$$

$$-\ln 2 + (T-1)\ln \lambda_1 + \ln\left(1 + \frac{\sqrt{b}}{2}\right) + \ln\left(1 + \frac{1}{1+\frac{\sqrt{b}}{2}}\left(\frac{\lambda_2}{\lambda_1}\right)^{T-1}\right) \leq$$

$$-\ln 2 + (T-1)\sqrt{b} + \frac{\sqrt{b}}{2} + \left(\frac{\lambda_2}{\lambda_1}\right)^{T-1} \ ,$$

where the last term is expanded in Lemma 5. Theorem 1 follows by substituting $b = \frac{B^2}{an}$.

6 Comments on the Proof

In this section we make some remarks about the proof.

Remark 1. Inequality (12) can be iterated, but that method would not lead to a good upper bound. For equal as, by using Stirling's formula we get

$$\ln \frac{\det A}{a^{nT}} \le \ln \prod_{t=1}^{T} \left(\frac{tB^2}{an} + 1 \right)^n \approx n \ln T! + Tn \ln \frac{B^2}{an} \approx Tn \ln T - Tn \ln a \frac{n}{B^2} \ .$$

In order to get an extra term of the order $o(T)$, we must take $a(T)$ growing at about the same rate as T and thus ruin the growth of $a \cdot \sum_{t=2}^{T} \|u_t - u_{t-1}\|^2$.

Remark 2. A recurrent formula upper bounding the determinant of $\overline{A}_2$ can be obtained in a simpler way not involving Proposition 5 at a price of a small loss of quality.

If the diagonal blocks $x_t x_t'/a + cI$ in $\overline{A}_2$ are replaced by $\left(\frac{B^2}{a} + c \right) I$, the eigenvalues and the determinant may only increase. Indeed, each matrix $\frac{B^2}{a} I - x_t x_t'/a$ is positive semidefinite and adding the positive semidefinite block diagonal matrix will not increase the eigenvalues by the Courant-Fischer min-max theorem ([HJ13], Theorem 4.2.6). The resulting matrix turns out to be the Kronecker (tensor) product of I and the tridiagonal $(T \times T)$-matrix

$$\breve{A}_2 = \begin{pmatrix} \frac{B^2}{a} + 2 & -1 & 0 & & \\ -1 & \frac{B^2}{a} + 2 & -1 & & \\ & \ddots & \ddots & \ddots & \\ & & -1 & \frac{B^2}{a} + 2 & -1 \\ & & 0 & -1 & \frac{B^2}{a} + 1 \end{pmatrix} \ .$$

Theorem 4.2.12 from [HJ94] on eigenvalues of the Kronecker product implies

$$\det \overline{A}_2 \le \det(I \otimes \breve{A}_2) = (\det I)^T (\det \breve{A}_2)^n = (\det \breve{A}_2)^n \ .$$

The determinant of $\breve{A}_2$ can be calculated using the recurrence from [HJ13], Section 0.9.10 (this is effectively a non-block special case of Proposition 4). We get an upper bound on $\det \overline{A}_2$ similar to Lemma 3 but with $b = B^2/a$.

Then using Lemmas 4 and 5 we get an analogue of Theorem 1 with a slightly different α (which is not important) and $Y^2 B (T - 1/2) \frac{n}{\sqrt{a}}$ instead of $Y^2 B (T - 1/2) \sqrt{\frac{n}{a}}$. Applying Lemma 1 we get a counterpart of Corollary 1 but with the main extra term $\frac{3}{2^{2/3}} Y^{4/3} B^{2/3} n^{2/3} (T - 1/2)^{2/3} (v(T))^{1/3}$.

Acknowledgement

The author has been supported by the Leverhulme Trust through the grant RPG-2013-047 'Online self-tuning learning algorithms for handling historical information'. The author would like to thank Vladimir Vovk, Dmitry Adamskiy, and Vladimir V'yugin for useful discussions. Special thanks to Alexey Chernov, who helped to simplify the statement of the main result.

References

AKCV12. D. Adamskiy, W. M. Koolen, A. Chernov, and V. Vovk. A closer look at adaptive regret. In *Algorithmic Learning Theory*, pages 290–304. Springer, 2012.

AW01. K. S. Azoury and M. K. Warmuth. Relative loss bounds for on-line density estimation with the exponential family of distributions. *Machine Learning*, 43:211–246, 2001.

BB61. E. F. Beckenbach and R. E. Bellman. *Inequalities*. Springer, 1961.

BK07a. S. Busuttil and Y. Kalnishkan. Online regression competitive with changing predictors. In *Algorithmic Learning Theory, 18th International Conference, Proceedings*, pages 181–195, 2007.

BK07b. S. Busuttil and Y. Kalnishkan. Weighted kernel regression for predicting changing dependencies. In *Machine Learning: ECML 2007*, pages 535–542. Springer, 2007.

CBCG05. N. Cesa-Bianchi, A. Conconi, and C. Gentile. A second-order perceptron algorithm. *SIAM Journal on Computing*, 34(3):640–668, 2005.

For99. J. Forster. On relative loss bounds in generalized linear regression. In *Fundamentals of Computation Theory*, pages 269–280. Springer, 1999.

HJ94. R. A. Horn and C. R. Johnson. *Topics in Matrix Analysis*. Cambridge University Press, 1994.

HJ13. R. A. Horn and C. R. Johnson. *Matrix analysis*. Cambridge University Press, 2nd edition, 2013.

HW01. M. Herbster and M. K. Warmuth. Tracking the best linear predictor. *Journal of Machine Learning Research*, 1:281–309, 2001.

MVC15. E. Moroshko, N. Vaits, and K. Crammer. Second-order non-stationary online learning for regression. *Journal of Machine Learning Research*, 16:1481–1517, 2015.

Sal06. D. K. Salkuyeh. Comments on "A note on a three-term recurrence for a tridiagonal matrix". *Applied mathematics and computation*, 176(2):442–444, 2006.

Vov90. V. Vovk. Aggregating strategies. In *Proceedings of the 3rd Annual Workshop on Computational Learning Theory*, pages 371–383, San Mateo, CA, 1990. Morgan Kaufmann.

Vov98. V. Vovk. A game of prediction with expert advice. *Journal of Computer and System Sciences*, 56:153–173, 1998.

Vov01. V. Vovk. Competitive on-line statistics. *International Statistical Review*, 69(2):213–248, 2001.